Algorithm for normal random numbers

Julio F. Fernández[1] and Carlos Criado[2]
[(1)] *Instituto de Ciencia de Materiales de Aragón*
Consejo Superior de Investigaciones Científicas
and Universidad de Zaragoza, 50009-Zaragoza, Spain
[(2)] *Departamento de Física Aplicada I, Universidad de Málaga*
29071-Málaga, Spain
(February 1, 2008)

We propose a simple algorithm for generating normally distributed pseudo random numbers. The algorithm simulates N molecules that exchange energy among themselves following a simple stochastic rule. We prove that the system is ergodic, and that a Maxwell like distribution that may be used as a source of normally distributed random deviates follows in the $N \to \infty$ limit. The algorithm passes various performance tests, including Monte Carlo simulation of a finite 2D Ising model using Wolff's algorithm. It only requires four simple lines of computer code, and is approximately ten times faster than the Box-Muller algorithm.

Pseudo random number (PRN) generation is a subject of considerable current interest [1]. Deterministic algorithms lead to undesirable correlations, and some of them have been shown to give rise to erroneous results for random walk simulations [2], Monte Carlo (MC) calculations [3,4], and growth models [5]. Most of the interest has been focused on PRN's with uniform distributions. Less attention has been paid to non-uniform PRN generation.

Sequences of random numbers with Gaussian probability distribution functions (pdf's) are needed to simulate on computers gaussian noise that is inherent to a wide variety of natural phenomena [6]. Their usefulness transcends physics. For instance, numerical simulations of economic systems that make use of so called *geometric* Brownian models (in which noise is multiplicative) also need a source of normally distributed PRN's [7]. There are several algorithms available for PRN's with Gaussian pdf's [8]. Some, such as Box-Muller's algorithm, require an input of uniform PRN's, and their output often suffers from the pitfalls of the latter [9]. Robustness is therefore a relevant issue. In addition, Box-Muller's algorithm is slow and can consequently consume significant fractions of computer simulation times [10]. The comparison method demands several uniform PRN's per normal PRN, and is therefore also slow [11]. Use of tables [12] is not a very accurate method. Algorithms that are related, but not equivalent, to the one we propose here have been published [10,13], but they are somewhat cumbersome to use. In addition, no proof of their validity has been given.

We propose here a new algorithm for the generation of normally distributed PRN's that is quite simple and fast. It is a stochastic caricature of a closed classical system of N particles. Their velocities provide a source of PRN's. We prove that, for any initial state, their pdf becomes Maxwellian in the $N \to \infty$ limit, after an infinite number of two-particle "collisions" take place. To this end, we first prove that our system is ergodic [14,15]. The proof is not exceedingly difficult because our system is not deterministic. We also study its output as a function of N, and establish useful criteria for its implementation. Correlation test results are also reported.

For the motivation, consider numbers $v_1, v_2 \dots v_N$, placed in N computer registers, analogous to velocities of N particles that make up a closed classical system in 1D. Pairs of registers $\imath$ and $\jmath$, say, selected at random without bias, are to "interact" somehow, conserving quantity $v_i^2 + v_j^2$. By analogy with the approach to equilibrium (i.e., to Maxwell's velocities distribution) that is believed to take place in Statistical Physics, we expect that sufficient number of iterations will lead to an approximately Gaussian pdf of register values, from which the desired PRN's may be drawn. (See also Ref. [10].) We define below the simplest interaction we can think of in order that (1) implementation on a computer be very fast, and (2) that we may be able to prove that a Gaussian pdf does indeed ensue.

Before the algorithm is implemented, all N registers must be initialized to, say, $v_\imath = 1$ for all $\imath$ satisfying $1 \le \imath \le N$, or all $v_\imath$ may be read from a set of N register values saved from a previous computer run, which we assume to fulfill $\sum v_i^2 = N$. Let $U(1,N)$, $U_\imath(1,N)$ be unbiased integer random variables, both in the interval $[1,N]$, except that $U_\imath$ cannot equal $\imath$. The algorithm follows:

$$\imath = U(1,N);\ \jmath = U_\imath(1,N); \tag{0.1}$$

$$v_\imath \leftarrow (v_\imath + v_\jmath)/\sqrt{2}; \tag{0.2}$$

$$v_\jmath \leftarrow -v_\imath + \sqrt{2} v_\jmath \tag{0.3}$$

The updated value of $v_\imath$, from Eq. (2), is used in Eq. (3). After an initial *warm up* phase (see below), $v_\imath$ and $v_\jmath$ may be drawn each time transformation (1-3) is applied. They are two independent PRN's, each one with an approximately Gaussian pdf, with $\langle v_\imath \rangle = 0$ and $\langle v_\imath^2 \rangle = 1$ for all $\imath$, if N is sufficiently large (see below). Transformation (1-3) may be thought of as a rotation of $\pm\pi/4$ with respect to a randomly chosen $\imath\jmath$ plane ($+$ and $-$ signs are for the two possible index orderings, $\imath\jmath$ and $\jmath\imath$). Thus, quantity $\sum v_\ell^2$ is conserved. Frequencies of events from sequences of 10^6, 10^8 and 10^{10} PRN's generated with transformation (1-3), with $N = 1024$, are exhibited in Fig.1.

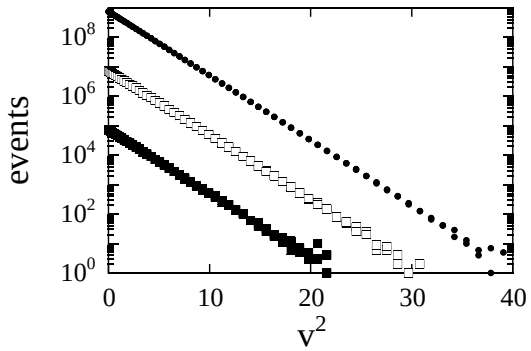

FIG. 1. Number $n(v)$ of PRN's generated within $v - \Delta v/2$ and $v + \Delta v/2$, for $\Delta v = 0.1$. Datapoints shown as $\bullet$, $\square$, and $\blacksquare$, follow from sequences of 1.9×10^{10}, 1.8×10^8, and 2×10^6 PRN's, respectively.

We first explain why PRN's genered by transformation (1-3) are expected to be normally distributed. Let $\mathbf{P}_n(\mathbf{v})$ be the probability density at $\mathbf{v} = (v_1, v_2, \dots, v_N)$, after transformation (1-3) has been applied n times, on the $(N-1)$-dimensional spherical surface $\mathcal{S}_{N-1}$, of radius $\sqrt{N}$ given by $N = \sum_{\ell=1,\dots,N} v_\ell^2$. Let the single register pdf $p(v)$ be the $n \to \infty$ limit of $p_n(v)$, where $p_n(v_1) = \int \mathbf{P}_n(\mathbf{v})\, dv_2 dv_3 \dots dv_N$. We show further below that $\mathbf{P}_n(\mathbf{v}) \to$ *constant* over $\mathcal{S}_{N-1}$ as $n \to \infty$. It then follows by integration that,

$$p(v) \propto \left(1 - \frac{v^2}{N}\right)^{(N-3)/2}. \tag{0.4}$$

Clearly, $p(v) \to C\exp(-v^2/2)$ in the $N \to \infty$ limit, which is the desired result.

We prove below, in three stages, that $P_n(\mathbf{v})$ does indeed become homogeneous over spherical surface $\mathcal{S}_{N-1}$, if $N \geq 3$, in the $n \to \infty$ limit. We first prove $P_n(\mathbf{v}) \leftrightarrow P_n(\mathbf{u})$ as $n \to \infty$ if $\mathbf{v}$ and $\mathbf{u}$ are related. [From here on, we say that points $\mathbf{v}$ and $\mathbf{u}$ are *related* if succesive transformations (1-3) of $\mathbf{v}$ can lead to $\mathbf{u}$.] We then prove that the system's "orbit" covers $\mathcal{S}_{N-1}$ densely [that is, that any point $\mathbf{v} \in \mathcal{S}_{N-1}$ can be brought arbitrarily close to any other point $\mathbf{u} \in \mathcal{S}_{N-1}$ by applying transformations (1-3) to $\mathbf{v}$ a sufficient number of times]. Then, the desired result follows easily. It may help to place the significance of the proof that follows into proper perspective to note that if in Eq. (1) $\jmath \leftarrow U_i[1, N]$ is replaced by $\jmath = \imath + 1 \; mod N$, the system becomes then non-ergodic, as can be easily checked numerically.

To start the proof, let kernel $K(\mathbf{v}, \mathbf{v}')$ be defined by $P_{n+1}(\mathbf{v}) = \int K(\mathbf{v}, \mathbf{v}') P_n(\mathbf{v}')\, d\mathbf{v}'$, and let

$$F_n \equiv \int \{P_{n+1}^2(\mathbf{v}) - P_n^2(\mathbf{v})\}\, d\mathbf{v}. \tag{0.5}$$

Note first that $F_n < 0$ implies that $P_{n+1}(\mathbf{v})$ is more uniform than $P_n(\mathbf{v})$, in the sense that $\int d\mathbf{v}\; [P_{n+1}(\mathbf{v}) - \overline{P}]^2 < \int d\mathbf{v}\; [P_n(\mathbf{v}) - \overline{P}]^2$, where $\overline{P} = 1/\int dv$. It follows from the definition of $K(\mathbf{v}, \mathbf{v}')$ that

$$F_n = \int d\mathbf{v} \{[\int d\mathbf{v_1} K(\mathbf{v}, \mathbf{v}_1) P_n(\mathbf{v}_1)]^2 - P_n^2(\mathbf{v})\}. \tag{0.6}$$

Making use of the detailed balance condition, $K(\mathbf{v}, \mathbf{v}') = K(\mathbf{v}', \mathbf{v})$, which our system satisfies, and the relation $\int\; d\mathbf{v}\; K(\mathbf{v}, \mathbf{v}') = 1$, Eq. (0.6) can be cast into,

$$F_n = -\frac{1}{2} \int d\mathbf{v} \int d\mathbf{v_1} \int d\mathbf{v_2}\; Q(\mathbf{v}, \mathbf{v}_1, \mathbf{v}_2), \tag{0.7}$$

where, $Q = K(\mathbf{v}, \mathbf{v}_1) K(\mathbf{v}, \mathbf{v}_2) [P_n(\mathbf{v}_1) - P_n(\mathbf{v}_2)]^2$. Therefore, in the $n \to \infty$ limit, $P_n(\mathbf{v})$ becomes constant over each set in $\mathcal{S}_{N-1}$ within which any two points $\mathbf{v}, \mathbf{u}$ are related.

We now prove that the system's orbit covers $\mathcal{S}_{N-1}$ densely. Let H_N be the group of transformations in N dimensions defined by Eqs. (1-3). We first show that any rotation in 3D [that is, any element of $SO(3)$] can be approximated arbitrarily close by elements of H_3. The proof is extended to higher dimensions by induction. Note first that H_3 does not belong to the set of *finite* rotation groups in 3D [16], and is therefore an infinite group. Let group $SO(3)$ be covered by spheres of radius $\epsilon/2$ each. A finite number of them is sufficient, since the volume of $SO(3)$ is finite [17]. It follows that there must be at least one sphere with two elements of H_3 in it, since H_3 has an infinite number of elements. Let these two elements be r and s, and let $g(\mathbf{u}, \epsilon)$ be element rs^{-1} of H_3, which is a rotation by angle ϵ about some undetermined $\mathbf{u}$ axis. We will build elements of H_3 that are as near as desired to any given rotation. To this end, it is sufficient to show that it can be done for a set of infinitesimal generators of rotations [18]. One such set is made up of infinitesimal rotations about three linearly independent axes. Consider axes $\mathbf{u}_1$, $\mathbf{u}_2$, and $\mathbf{u}_3$ that are obtained from $\mathbf{u}$ by rotations $g(1, \pi/2)$, $g(2, \pi/2)$, and $g(3, \pi/2)$ about each one of the coordinate axes by angle $\pi/2$. The correspondng infinitesimal rotations are given by [17], $g(\mathbf{u}_\imath, \epsilon) = g(\imath, \pi/2) g(\mathbf{u}, \epsilon) g^{-1}(\imath, \pi/2)$. This concludes the proof for 3 dimensions.

We now prove by induction that any element $g(\imath\jmath, \alpha)$, for the rotation about plane $\imath\jmath$, by angle α of the rotation group $SO(N)$ can be approximated as nearly as desired by an element g of group H_N, for $N > 3$. By hypothesis, any $g(\imath\jmath, \alpha)$, for $\imath, \jmath = 1, 2, \ldots, N$ can be approximated by an element g of H_N. We show now that $g(\imath\, N+1, \alpha)$, for $\imath = 1, 2, \ldots, N$, can also be approximated by elements of H_{N+1}. We take $g \in H_N$ within distance ϵ of $g_{\imath\jmath}(\alpha)$. Now, since rotations preserve distances, it follows that $g(\imath\; N+1, \alpha) \in SO(N+1)$, given by $g(\imath\; N+1, \alpha) = g(\imath\, N+1, \pi/2) g(\imath\jmath, \alpha) g^{-1}(\imath\, N+1, \pi/2)$ is within distance ϵ of $g' \in H_{N+1}$, given by $g' = g(\imath\, N+1, \pi/2) g g^{-1}(\imath\, N+1, \pi/2)$. This proves dense coverage in $N \geq 3$ dimensions. This is a stochastic generalization of Jacobi's theorem [15] to more than two dimensions.

To conclude the proof that $P_n(\mathbf{v}) \to$ *constant* in the $n \to \infty$ limit, consider any two points $\mathbf{V}$ and $\mathbf{U}'$ as centers of disks $\mathcal{D}_{\mathbf{V}}$ and $\mathcal{D}_{\mathbf{U}'}$, both of radius r, in $\mathcal{S}_{N-1}$. Since the system's orbit covers $\mathcal{S}_{N-1}$ densely for $N \geq 3$, it follows that a point $\mathbf{U}$ that is related to $\mathbf{V}$ exists arbitrarily close to $\mathbf{U}'$. Consider now disks $\mathcal{D}_{\mathbf{V}}$ and $\mathcal{D}_{\mathbf{U}}$. The fact that there exists at least one sequence of rotations in H_N that take $\mathbf{V}$ into $\mathbf{U}$ implies that there exists at least one single rotation g in H_N that transforms $\mathbf{V}$ into $\mathbf{U}$. Since g is a rotation, it transforms $\mathcal{D}_{\mathbf{V}}$ *rigidly* into $\mathcal{D}_{\mathbf{U}}$. It follows that $\int d\mathbf{v}\; P_N(\mathbf{v})$ over $\mathcal{D}_{\mathbf{V}}$ equals $\int d\mathbf{u}\; P_N(\mathbf{u})$ over $\mathcal{D}_{\mathbf{U}}$. Since r is arbitrary, and $\mathbf{V}$ and $\mathbf{U}'$ are any two points in $\mathcal{S}_{N-1}$, it follows that $P(\mathbf{v})$ is constant over $\mathcal{S}_{N-1}$ (except, perhaps, on a set of measure zero). This is the desired result. Ergodicity follows [15].

We next address the following practical issues: (1) how good an approximation to a Gaussian pdf of PRN's is achieved with a necessarily *finite* set of N registers; (2) how long must the warm up phase be.

It is convenient to rewrite Eq. (0.4) as follows,

$$p(v) \propto e^{-v^2/2} e^{g_N(v)/N}. \tag{0.8}$$

where $g_N(v) = v^2(3 - v^2/2)/2 + \mathcal{O}(1/N)$. $N^{-1} g_N(v)$ is approximately the fractional deviation, $\delta p(v)/p(v)$, from Gaussian form if $\delta p(v)/p(v) \ll 1$. We have checked this behavior numerically. Clearly, the *number of registers* N that must be used increases with the number M of PRN's one intends to generate. This is because the value of the largest PRN generated increases, on the average, with M. More precisely, the value of v beyond which PRN's are only generated with probability q is approximately given by $v^2 \approx 2\ln(M/vq)$. Now, it follows from Eq. (0.8) that the fractional error $\delta P/P$ in the probabil-

ity density at v is approximately $N^{-1}v^2(3-v^2/2)/2$ for very large N. (It is pointless to require this error to be too small since a PRN is expected to be generated beyond x with a small probability q.) It then follows that $[\ln(M/qv)]^2 \lesssim N\delta P/P$ must be satisfied by N. Thus, approximately 10^4 registers are sufficient in order to generate as many as 10^{15} PRN's, with a roughly 10% error in the probability for the *largest* PRN in the sequence. For results obtained from a sequence of 10^{10} PRN's generated with 1024 registers, see Fig. 1.

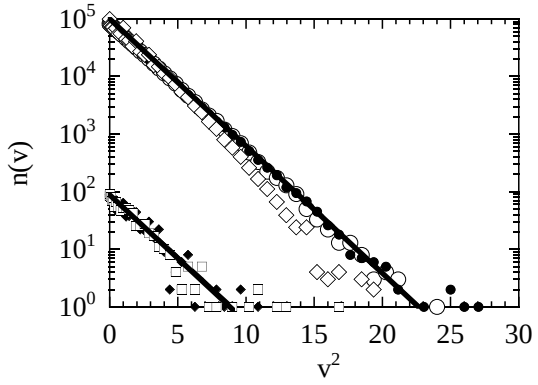

FIG. 2. Number $n(v)$ of PRN's within $v-\Delta v/2$ and $v+\Delta v/2$, for $\Delta v = 0.1$, starting from initial conditions $v_\imath = 1$, for all $\imath \in [1, N]$, after transformation (1-3) is iterated $2n_pN$ times (that is, after each register interacts, on the average, $2n_p$ times). The □ and ♦ stand for $n_p = 2, 10$, respectively, for $N = 1024$. The ◇ and •, and ∘ stand for $n_p = 2, 4$, and 10 respectively, for $N = 1048576$. The two straight lines stand for $C\exp(-v^2/2)$ for two values of C.

Our algorithm must be applied a number n_pN of times before it is ready for use unless all $v_\imath$ are initialized with"equilibrium" values (stored from some previous computer run). The distribution of all register values then evolves towards equilibrium, as illustrated in Fig. 2. Deviations from equilibrium are statistically insignificant for $n_p \gtrsim 2$ and $N = 1024$, and for $n_p \gtrsim 4$ and $N = 1\,048\,576$. Since n_p is expected to increase as $\ln N$, $n_p = 8$ should provide ample warm up for any forseeable applications.

The number of PRN's that must be generated before each PRN in sequence $v_1, v_2, \ldots, v_N$ returns within distance r from its initial value is exponential in N. More specifically, we estimate it to be $(\tau/\sqrt{N})(1/r)^N$ for $N \gg 1$, where τ is the period of the algorithm used to select $\imath$ and $\jmath$ in Eq. (1). The estimation is based on $P_n(\mathbf{v}) \to$ *constant* over $\mathcal{S}_{N-1}$ as $n \to \infty$. Thus, an effectively infinite recurrence time follows for any reasonable value of N.

Correlations between a finite number of PRN's clearly vanish as $N \to \infty$, since $\imath$ and $\jmath$ in Eq. (1) are supposedly independent PRN's. We have searched for correlations in m succesively generated PRN's $v_1, v_2, \ldots v_m$, for $m = 3, 4, \ldots, 6$, performing a chi-square isotropy test over the corresponding m-dimensional space. An m-tuple $\mathbf{v} = v_1, v_2, \ldots, v_m$ was said to belong to the $i-th$ cone, of 1024 randomly oriented cones with axes $\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{1024}$, if $0.99 \le \mathbf{v}.\mathbf{w}_\imath \le 1$. No significant deviations from isotropy were observed for 10^6 generated m-tuples.

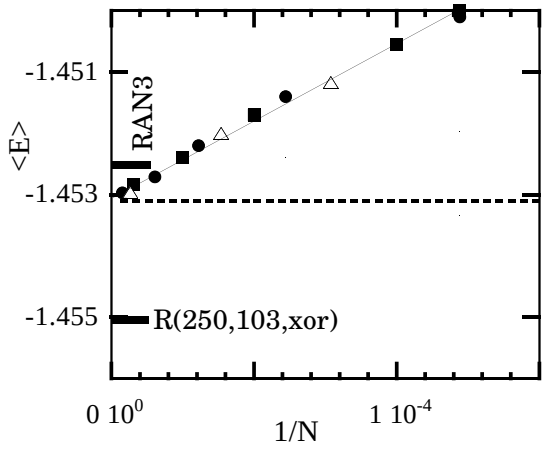

FIG. 3. Average energy per spin, obtained from MC simulations using Wolff's algorithm, versus the inverse of the number of registers used for the generation of PRN's with Gaussian pdf's. •, ■, and △ stand for data points that follow from feeding our algorithm with the following uniform PRN generators: ggl, R(250,103,xor), and Ran3, respectively. Unacceptable energy values that have been obtained in Refs. [3] using R(250,103,xor), and Ran3 are also shown as bars next to the y-axis.

Implementation of Wolff's algorithm [19] in MC calculations of the Ising model's critical behavior is a demanding test that some well known uniform PRN generators have failed [3]. Large clusters are then flipped as a whole, and this tests correlations in very long sequences. We have used normal PRN's generated by our algorithm as input into a MC simulation of an Ising system of 16×16 spins at the critical temperature. [For that, we note that $v_\imath^2 + v_\jmath^2 > 2x$ as often as $u > \exp(-x)$ if $v_\imath$ and $v_\jmath$ (u) are PRN's with Gaussian (uniform) pdf's, respectively.] The energy obtained is shown in Fig. 3 as a function of the number of registers N. The following uniform PRN algorithms were used to select $\imath$ and $\jmath$ in Eq. (1): ggl [3], R(250, 103,xor) [2,3], and RAN3 [20]. We tried the latter two algorithms, which have been shown to lead by themselves to unacceptable results for the Ising model [3], in order to test our algorithm's robustness. The results shown in Fig. 3 are gratifying.

Similarly, the specific heat c and magnetization m fluctuations data points obtained follow approximately the relations $c \simeq c_0 + 8.4/N$, and $\langle(\delta m)^2\rangle \simeq \chi_0 + 33/N$, respectively, where $c_0 = 1.497(1)$ and $\chi_0 = 0.5454(2)$, in agreement with the known exact values [3,21].

Double precision is recommended. It prevents excesive drift of the sum $\sum v_i^2$ away from its assigned value. Even then, single precision accuracy is to be expected at the end of a sequence of some 10^{16} PRN's, unless the sum is normalized several times during the run.

In summary, we have shown that implementation of Eqs. (1-3) provides a source of PRN's with an approximately Gaussian pdf. Some 10^4 registers (molecules) are sufficient for some purposes, but up to 10^5 or more may be necessary for more demanding tasks. (Having to make a decision about the number of registers to be used may sometimes be an unwelcomed task. On the other hand, it is a virtue of the algorithm, that one can control, through the value of N, how close the output is to be from sequences of truly independent random numbers with Gaussian pdf's.) Initial warm ups for arbitrary initial conditions are necessary; it is sufficient to let each register initially interact an average number of, say, 8 times. The system's recurrence time was shown to be exponential in N, and therefore effectively infinite. Its behavior appears to be robust. The proposed algorithm runs an order of magnitude faster on computers than the most often used Box-Muller method [8,9]. For a fortran code of our algorithm or other questions, please write JFF@Pipe.Unizar.Es.

Continuous help from Dr. Pedro Martínez with computer systems is deeply appreciated by JFF. We are indebted to Prof. P. Grassberger for an important suggestion. JFF and CC are grateful for partial financial support from DGICYT of Spain, through grants No. PB95-0797 and PB97-1080, respectively.

[1] For a general introduction to generation of uniform PRN sequences, see, D.E. Knuth, *The Art of Computer Programming*, Vol. 2, second edition (Addison-Wesley, Reading, MA, 1981); G. Marsaglia, in *Computer Science and Statistics: The Interface*, edited by L. Billard (Elsevier-North Holland, Amsterdam, 1985); P. L'Ecuyer, Ann. Oper. Res. **53**, 77 (1994); H. Niederreiter, J. Comp. Appl. Math. **56**, 159 (1994).

[2] P. Grassberger, Phys. Lett A **181**, 43 (1993); L. N. Shchur, J. R. Heringa, and H. W. J. Blöte, Physica A, **241**, 579 (1997).

[3] A. M. Ferrenberg, D. P. Landau, and Y. J. Wong, Phys. Rev. Lett. **69**, 3382 (1992); I. Vattulainen, T. Ala-Nissila, and K. Kankaala Phys. Rev. Lett. **73**, 2513 (1994); F. James, Europhys. Lett., **24**, 24 (1993); L. N. Shchur and H. W. J. Blöte, Phys. Rev. B, **55**, R4905 (1997).

[4] F. J. Resende and B. V. Costa, Phys. Rev. E **58**, 5183 (1998).

[5] R. M. D'Souza, Y. Bar-Yam, and M. Kardar, Phys. Rev. E, **57**, 5044 (1998).

[6] For old work, see N. Wax, *Noise and Stochastic Processes* (Dover Publications Inc., 1954); for a more recent general account, see, C.W. Gardiner, *Handbook of Stochastic Methods* (Springer-Verlag, Berlin, 1990).

[7] A. R. Bergstrom, Econometric Theory **13**, 467 (1997); C. Rose, J. of Economic Dynamics and Control **19**, 1391 (1997).

[8] See for instance, Devroye, *Non-Uniform Random Variate Generation* (Springer-Verlag, New York, 1986); D.E. Knuth, *The Art of Computer Programming*, Vol. 2, second edition (Addison-Wesley, Reading, MA, 1981) p. 117.

[9] See for instance, P. Bratley, B. L. Fox and L. E. Schrage, *A guide to Simulation* (Springer-Verlag, New York, 1983) pp. 223-224.

[10] J. F. Fernández and J. Rivero, Comput. Phys. **10**, 83 (1996).

[11] G.E. Forsythe, Math. of Comp. **26**, 817, (1972).

[12] See, for instance, R. Toral and A. Chakrabarti, *Comput. Phys. Commun.* **74**, 327 (1993) and references therein.

[13] C. S. Wallace, ACM Trans. Math. Software **22**, 119 (1996).

[14] See, for instance, J. Lebowitz and O. Penrose, Physics Today **26**, 23 (1973).

[15] V. I. Arnold and A. Avez, *Ergodic Problems of Classical Mechanics* (W. A. Benjamin, Inc., New York, 1968).

[16] H. S. M. Coxeter *Introduction to Geometry* (John Wiley and Sons, New York, 1969); A. I. Kostrinkin, *Introduction to Algebra* (MacGraw-Hill, New York, 1992).

[17] See, for instance, R. Gilmore, *Lie groups, Lie algebras, and some of their applications* (John Wiley and Sons, New York, 1974). p. 209.

[18] H. Goldstein, *Classical Mechanics* (Addison-Wesley, Reading, 1959) pp. 124-132.

[19] U. Wolff, Phys. Rev. Lett. **62**, 361 (1989).

[20] W. H. Press, B. P. Flannery, S.A. Teukolsky, and W. T. Vetterling, *Numerical Recipes* (Cambridge University Press, Cambridge, UK, 1989).

[21] A. E. Ferdinand and M. E. Fisher, Phys. Rev. **185**, 832 (1969).